

# for

`for` loops take an iterator variable and assign it successive values from a sequence or vector. `for` loops are most commonly used for iterating over the elements of an object (list, vector, etc.)

```
for(i in 1:10) {  
    print(i)  
}
```

This loop takes the `i` variable and in each iteration of the loop gives it values 1, 2, 3, ..., 10, and then exits.

# for

These three loops have the same behavior.

```
x <- c("a", "b", "c", "d")

for(i in 1:4) {
  print(x[i])
}

for(i in seq_along(x)) {
  print(x[i])
}

for(letter in x) {
  print(letter)
}

for(i in 1:4) print(x[i])
```

# Nested for loops

for loops can be nested.

```
x <- matrix(1:6, 2, 3)

for(i in seq_len(nrow(x))) {
  for(j in seq_len(ncol(x))) {
    print(x[i, j])
  }
}
```

Be careful with nesting though. Nesting beyond 2–3 levels is often very difficult to read/understand.